IN THE SPECIFICATION:

Please amend the specification by substituting the following replacement paragraphs:

On Page 5, please amend the **Brief Description of the Drawings** as follows:

Further characteristics, features, and advantages of the present invention will be apparent upon consideration of the following detailed description of the invention, taken in conjunction with the following drawings, and in which:

Fig. 1 is a block diagram illustrating an automatic exchange according to an embodiment of the invention interconnected with a plurality of market makers, exchange members, and other entities;

Fig. 2 is a detailed block diagram illustrating the exchange of Fig. 1;

Figs. 3 through 13 are flow charts showing processing by the exchange of Fig. 1[.] ; and

Fig. 14 is a flow chart showing processing of a complex order according to an embodiment of the invention.

On Page 14, please amend the third full paragraph as follows:

The final two ~~process~~ processes in Fig. 2 are the tick-worse process 39 and the step-up or ~~tick-worse~~ tick-down process ~~39~~ 38, both of which apply to quotations entered by a PMM 3 or CMMs 5, 7. The tick-worse process 39 automatically changes the price and size of a quotation in the book memory 33 when the size of the quotation in the book memory 33 is reduced to zero. The tick-worse process 39 determines the new price and size according to parameters entered by the PMM 3 or CMM 5, 7. These parameters are stored in the system memory 26 with respect to each quotation entered by a PMM 3 or CMM 5, 7. The new quotation price is one or more trading increments lower for a quotation to buy and one or more trading increments higher for a quotation to sell. The step-up or tick-down process 38 increases the size of a quotation at the best price when the aggregate size of the best price would be less than the minimum market size according to parameters entered by the PMM 3 or CMM 5, 7, or sends the quotation to the tick-worse process 39. The tick-worse process 39 and the step-up or tick-down process 38 are described in detail below.

On page 16 please amend the first full paragraph as follows:

If the incoming order is an order to sell at step S27, the order process 25 determines ~~att~~ <u>at</u> step S41 of Fig. 3(c) whether the incoming order can trade with the bid side of the book memory 33. If the order cannot trade at step S41, the order process 25 determines at step S43(a) whether the order would improve upon the best offer, that is, whether the price of the order is lower than the lowest order or quotation stored in the book memory 33, and if it is not, the order process 25 stores the order in the book memory 33 at step S43(b). If the order would improve upon the best offer, the order process 25 determines at step S43(c) whether the size of the order is equal to or greater than the minimum market size stored in the system memory 26, and if it is, stores the order in the book memory 33 at step S43(d). If the order is less than the minimum market size, the order process 25 determines at step S43(e) whether the order is a public customer order, and if it is not, deletes the order at step S43(f). If the order is a public customer order the order process 25 sends the order to the derive or trade process 32 at step S43(g).

On page 22, please amend the second full paragraph as follows:

As a fifth example, assume a public customer limit order to sell 49 contracts at 3 is entered. The offer matching process 36 proceeds as shown in Figs. 5(a) and 5(b). The offer matching process 36 determines at step S200 that there is a public customer order at the highest bid and matches 3 contracts of the incoming order against the ~~CUS~~ <u>public customer</u> order at step S216. The offer matching process moves to step S218 and determines that there remains 46 contracts in the incoming order, then moves to step S202 to determine that the original size of the order is greater than 5 contracts, which is the PMM small order preference size for the purposes of this example. The offer matching process 36 then applies the allocation formula and completes steps S222, S226, S230 and S234 in Fig. 5(b) as explained in previous examples, to determine that there are more than two professionals with orders or quotations along with the PMM 3 quotation at the highest bid price. Step S238 allocates the remaining 46 contracts among the PMM 3 and the three professionals. According to one embodiment, the order is allocated according to the following formula:

On page 24 please amend the second full paragraph as follows:

In this example, the PMM 3 has placed an initial quotation for 30 contracts shown in the first line of Table IV and has specified a range of quotation sizes to be quoted when the initial quotation is exhausted. The table is stored in the system memory 26 and is retrieved by the tick-worse process 39 when needed. The depth of the table and the number of contracts for each tick level can be adjusted by the PMM 3. When the initial quotation for 30 contracts, as shown in the first line of Table IV, is exhausted, the tick-worse process 39 automatically generates a new quotation for 10 contracts at a price one tick worse than the market price. If the initial order for 30 contracts were placed at a market price of 3, then after these contracts are traded, the tick-worse process 29 39 automatically generates a new quotation for 10 contracts at 2 15/16 (assuming that below 3 the minimum trading increment is 1/16). When the market at the price one tick worse becomes exhausted, the tick-worse process 39 enters a quotation for 30 contracts at a price two ticks worse than the original quotation and so on.

On page 26, please rewrite the heading for the third paragraph as follows:

Step-Up or Tick-Worse Process

On page 28, please amend the first full paragraph as follows:

The exchange 1 according to one embodiment of the present invention maintains a minimum size at the best bid and best offer, referred to herein as X. The value of X is variable and may be change changed in the system memory 26. According to one embodiment of the invention, the minimum market size X is 10 contracts. When an incoming public customer limit order that cannot trade improves the market (that is, when an order to buy at a price higher than the best bid in the book memory 33 or an offer to sell at a price lower than the best offer in the book memory 33) and the size of the order is less than 10 contracts, the order process 25 sends the order to the derive or trade process 32 at step S43(g) of Fig. 3(c) if it is an order to sell and at step S31(g) of Fig. 3(b) if it is an order to buy. If an incoming professional order or quotation improves the best price for less than 10 contracts, it is deleted according to steps S43(f) of Fig. 3(c) and S31(f) of Fig. 3(b).

00178466

On page 30, please amend the fourth full paragraph as follows:

The derive or trade process 32 will also be initiated if <u>size at</u> the best price ~~become~~ <u>becomes</u> less than 10 contracts. For example, again assume the market illustrated in Table VIII, if the PRO order is cancelled and the PMM 3 moves its quotation to 2 ¼, the bid at 3 would be reduced to only the public customer order for 3 contracts. The derive or trade process 32 would be initiated by the order process, and an order would be derived for 7 contracts in step S333 as described in the previous example. Moreover, if a professional order for fewer than 10 contracts became the best bid in a similar manner to this example, the order would be automatically deleted. The step-up or tick-worse process 38 would be initiated if a quotation for fewer than 10 contracts were to be the become the best bid, as described above.

On page 31, please amend the third full paragraph as follows:

The away market matching table determines a matching size that is the number of contracts that the PMM 3 is willing to execute at the better price equal to the away market for a range of price differentials between the PMM's quotation and the away market 17<u>.</u> According to one embodiment of the invention, the away market matching table is arranged so that the PMM 3 is more likely to change his price to execute a public customer order at the better away market price where the price differential between the PMM's quotation and the away market 17 is small. Where a large price differential exists, the PMM 3 will only trade small orders or will decline to match automatically. At step S344 the volume of the incoming order is compared with the maximum volume for the price differential given in the away market matching table. If the volume of the order is less than ~~tae~~ <u>the</u> volume of the order shown in the away market matching table, then step S348 automatically trades the order against the PMM 3 at the away market price. If the volume of the incoming order is greater than the volume given by the away market matching table, then step S346 alerts the PMM 3 of the price differential between the exchange 1 and the away market 17. The PMM 3 can then decide whether to trade the incoming order at the away market price.

On page 32, please amend the first, second, and third full paragraphs as follows:

If an order is not traded automatically by the PMM 3 in step S348, the order is stored in the book memory 33, but is not displayed as the best price. Rather, the order is hidden, but

remains available for execution. For example, if the best bid in the book memory 33 is 4 and the best offer is 4 ¼, and the best bid from another market is ~~4 1/8~~ 4 ⅛, a public customer order to sell at 4 that is stored at step S346 may be executed against an incoming market order or limit order to buy at 4 ¼, or higher. In this example, both the public customer order to sell at 4 and the order to buy received a better price.

Fast Market Process

When the market for an options contract becomes highly volatile or when the rate at which orders are received becomes too great, it is possible that the best price in the book memory 33 will not accurately reflect the true price of the market. In such situations a fast market can be initiated by the entity administering the exchange 1 by setting a fast market parameter in the system memory 26. The exchange 1 also may monitor information received from the reporting entity 19 and automatically initiate a fast market condition ~~is~~ when such a condition is indicated for the security underlying an option.

As shown in Fig. 3(a), the order process 25 checks whether a fast market condition exists at steps S9, S17 and S23 by checking the value of the fast market parameter stored in the system memory 26. If a fast market condition exists, the incoming order is rejected at step S11 if it is a block order or a facilitation order, and at step S19 if it is a FARMM order, otherwise the order is sent to the fast market process 37. The fast market process 37 accumulates orders for a time period determined by the value of the fast market parameter. At the end of this time period, a trade is executed at a price calculated to clear a maximum number of orders at a single price. After the trade, incoming orders are once again accumulated for the time period and again trade at the end of the time period. This process provides an equitable price for market orders by preventing orders received within ~~aa~~ a short period of time from being traded at varying prices. The delay introduced by the fast market process also serves to dampen price fluctuations. Table XII shows an example of a range of delay time periods that can be set depending on the degree of volatility in the fast market according to one embodiment of the invention. Depending on the level of price volatility and/or trading volume, the fast market level can be adjusted to provide the desired degree of damping.

Please amend the first full paragraph, and the second paragraph on pages 33 to 34 as follows:

The operation of the fast market process 37 is illustrated in Figs. 10(a) through 10(d). Step S567 shown in Fig. 10(a) selects the best (highest) bid in the book memory 33 as the current bid. Step S569 calculates the number of contracts that would trade at the current bid. Step S571 determines whether the current bid is equal to the PMM's bid. If not, step S573 selects the next lower bid and begins step S569 again. If the PMM's bid has been reached, step S571 moves the process to step S577 in Fig 10(b), which selects the best (lowest) offer in the book memory 33 as the current offer. Step S578 calculates the number of contracts that would trade at the current offer. Step ~~S79~~ S579 determines whether the current offer is equal to the PMM's offer. If not, step S580 selects the next higher bid and begins step S578 again. If the PMM's offer has been reached, step S579 moves the process to step S585 in Fig. 10(c).

If there is only one price that maximizes the number of contracts that can be traded, step S585 trades orders at that price. If both the current bid and current offer prices will lead to the same maximum number of contracts traded, then step S589 determines whether the spread between the current bid and offer prices is an even number of ticks. If it is, orders are traded at the average of the current bid and current offer prices at step S591. If the spread is not even in step S589, the process stores a variable N, which equals the number of ticks (i.e., trading increments) in the spread minus two, and moves to step S595. If the spread is one tick wide in step S593, the process determines whether ~~thee~~ the instrument is a put option or a call option. If it is a put option, step S600 trades the orders at the lower ~~off~~ of the prices determined in S585. If it is a call option, step S599 trades the orders at the higher of the prices determined in Step S585.

On page 34, please amend the second full paragraph as follows:

The opening process 40 is initiated by the PMM 3 to trade orders and quotations accumulated when the exchange 1 is not executing trades, e.g., overnight. The opening process 40 employs the same process as the fast market process 37 described above and illustrated in Figs. 10(a) through 10(d) with one modification. Prior to moving to steps S587 or S589 of Fig. 10(c), the opening process 40 determines whether there would be any market or marketable limit order left unexecuted at the maximum price or prices. If there would not be, the opening process 40 moves to steps S587 or S589, and if the there would be unexecuted market orders, the opening process moves back to step S567.

On page 36, please amend the first full paragraph as follows:

The allocation algorithm described in Fig. 4(b) is applied at steps S667 and S687 and the allocation algorithm described in Fig. 5(b) is applied in steps S671 and S694. For the purposes of the allocation algorithm illustrated in Figs. 4(b) and 5(b), responses received in step S656 are treated the same as professional orders and quotations.

On page 37, please amend the first full paragraph as follows:

Table XIV shows that there is are a total of 520 contracts available to match against the block order to buy. In step S662, it is determined that the order is to buy. In step S664, the price of the transaction is determined. In this example, the block order was to buy 500 at 2 ½. Since there are more than 500 contracts to match at 2 ½ and no responses or orders and quotations in the book memory 33 at a price lower than the block order price, the execution price is determined to be 2 ½, and there is nothing to execute at step S665. The public customer order for 10 contracts is executed at step S666, and the block order and facilitation process 35 applies the allocation algorithm in step S667 according to the same process illustrated in Fig. 4(b).

On page 38, please amend the first and second full paragraphs and the third paragraph continuing from page 38 to page 39 as follows:

As illustrated in Fig. 11(c), the process for execution of facilitation orders trades a defined certain percentage of the original size of the facilitation against to the EAM 9, 11 in steps S6S5 and S693 that entered the facilitation order prior to applying the allocation algorithms illustrated in Figs. 4(b) and 5(b) in steps S687 and S696 respectively. Further, after completing steps S687 and S696, any remaining unexecuted portion of the facilitation order is traded against the EAM 9, 11 that entered the facilitation order. The percentage of the facilitation order automatically executed against the EAM 9, 11 that entered the facilitation is a value that is stored in the system memory 26.

Quotation Matching Process

Fig. 13 illustrates the quotation matching process 31, which introduces a delay before automatically matching a bid and an ask quotation. The idea is to prevent quotations from matching only because one participant's automatic quotation system updates its quotation

slightly more quickly than another participant's automatic quotation system. As an example, if there is a quotation in the book memory 33 to buy at 4 and a CMM 7, 5 enters a quotation to sell at 4, the quotation matching process marks the quotation to buy and the quotation to sell in step S702 and then waits T seconds in step S704, T ~~beingg~~ being a variable stored in the system memory 26. In step S706, the process determines whether there exists a bid and an offer that match, since during T quotations may have ~~changes~~ changed. If there are no matching bids and offers, the marks from step S702 are removed in step S708. If there are bids and offers that match, the process marks those quotations in step S710 and executes the matching quotations that have two or more marks at step S712. The process then returns to step S704 to again wait T seconds.

COMPLEX ORDER PROCESS

A complex, combination, or spread order (complex order) contains a set of legs, each leg representing a different financial instrument on the same underlying product. Embodiments of the present invention described and illustrated herein are in terms of each of the legs consisting of series of equity options contracts. However, the invention is not limited to trading equity options complex orders. Rather, the invention can be implemented in any asset class or among several asset classes (such as, for example, legs consisting of index options contracts, futures contracts, stock, bonds, treasury instruments, exchange traded funds, security futures, etc. or any combination thereof), and a discussion of how the invention can be implemented in different or across asset classes is described in detail below.

On page 39, please amend the second paragraph starting on that page and extending onto page 40 as follows:

According to one embodiment of the invention, a centralized orderbook memory ~~to store~~ stores complex orders and regular orders. This orderbook may comprise the book memory 33, shown in Fig. 2. The complex order process 50 continually monitors the best price and size of all quotes and orders in the orderbook (including both complex orders and regular orders) (referred to as the best bid and offer or BBO), as well as market information from other linked markets (referred to as the OPRA BBO), to determine whether complex orders can match against other complex orders and regular orders. The complex order process 50 assigns priority levels to

complex orders and regular orders, and determines the best possible price to trade complex orders, whether it is against other complex orders or regular orders. The complex order process 50 ranks complex orders in time and price priority and matches complex orders according to this ranking to prevent complex orders to trade through or ahead of orders ranked higher in priority or orders at better prices. After receiving a complex order, the complex order process 50 displays the complex order to market participants for a ~~programable~~ programmable period to allow those participants to submit matching orders. Matching orders are ranked during this period in terms of price, allowing for price improvement in trading the complex order.

On page 41, please rewrite the last paragraph, which continues onto page 42 as follows:

Fig. 14 shows the operation of the complex order process 50 according to an embodiment of the invention. At Step ~~S-801~~ S801, a complex order is entered into the system by a PMM, CMM, or EAM. At Step ~~S-802~~ S802, the system validates the complex order. A valid complex order has a net price (which may be specified as fixed price or market price) and at least one leg. Each leg includes: (1) the series; (2) the ratio; (3) buy or sell value; and (4) the quantity. If the complex order is not valid, the complex order is rejected. If the complex order is valid, the complex order is sent to the orderbook and displayed to market participants. At Step ~~S-803~~ S803, the system introduces an optional delay of n seconds before the processor starts seeking matches for the complex order. The purpose of the delay is to provide market participants with an opportunity to trade against the order and/or improve the execution price of the order through submitting inverse (or opposite) orders. This parameter can be changed dynamically intraday by the administering exchange and can optionally be set to zero. If the complex order is traded during delay period, the complex order is removed from the ~~book~~ orderbook and reported to the relevant market participants.

On page 42, please amend the first full paragraph as follows:

If the complex order is not traded during the delay period, at Step ~~S-804~~ S804, the complex order process 50 is triggered and it checks the BBO for each leg of the complex order. The complex order process 50 rechecks the BBO each time it is retriggered. The complex order process 50 is retriggered by certain events, such as, for example: (1) a series status change; (2) a BBO update; (3) an OPRA BBO update; ~~and~~ or (4) a complex order update. When the complex

order process 50 is triggered, it: (a) selects the complex orders relating to the complex order or series that caused the trigger from the orderbook; (b) lists the selected complex orders by receipt time; (c) groups the selected complex orders by identical leg characteristics; and (d) sorts the selected complex orders with identical leg characteristics in price-time priority.

On page 43, please amend the second full paragraph as follows:

If the complex order is not marketable, the complex order is not traded and the system loops back to the monitoring the BBO at Step S~~-804~~ S804 and waits for the complex order process 50 to retrigger. If the complex order is marketable, the complex order process 50 determines whether an inverse complex order or regular orders on the orderbook offers a better price to trade the complex order at Step S~~-806~~ S806. To do so, the complex order process 50 (1) sorts the inverse complex orders by price then time priority, ignoring such orders if (a) they are less than the net price that was requested by the complex order, ~~and~~ or (b) their net price is outside of the valid spread range of the complex order; and (2) checks the best complex order price against the current best matching price against regular orders.

On page 45, please rewrite the first and second full paragraphs as follows:

As a result of the foregoing steps, at Step S~~-806~~ S806, the complex order process 50 compares the best inverse complex order price to the best regular order price. If regular orders represent the best price, the processor seeks to match the complex order against the regular orders. If an inverse complex order represents the best price, the complex order process 50 seeks to match the complex order against an inverse complex order. If both an inverse complex order and regular orders represent the same price and there would be no priority or other trading violations, the complex order process 50 seeks to match the complex order against an inverse complex order.

Specifically, at Step S~~-807~~ S807, if the complex order process 50 seeks to match the complex order with regular orders, the complex order process 50 checks whether any of the legs of the complex order are inverted. If none of the legs of the complex order are inverted, the complex order process 50 matches the trade against the regular orders at Step S~~-809~~ S809 according to the allocation algorithm described in figures 4a –5b. If one or more of the legs of

the complex order is inverted, the complex order process 50 seeks to match the complex order against an inverse complex order at Step ~~S 808~~ S808.

On page 46, please amend the third and fourth full paragraphs as follows:

After the complex order process 50 matches the complex order with the best inverse order, it executes the trade. The trade may consist of a trade between two complex orders on the orderbook at a specified net price; a complex order against regular orders on the orderbook at the price of the ~~book~~ orderbook, up to the limit of the complex order; and a complex order against regular orders on the orderbook, then trade the complex order against other complex orders on the orderbook.

After the complex order process 50 matches the complex order with the best regular orders, it executes the trade according to the allocation algorithm set forth in figures 4a-5b. Each leg of the complex order is executed at the ~~book~~ orderbook price, and the complex order process 50 ensures that the current BBO for each leg will not exceed the limit price of the complex order.

On page 47, please amend the second full paragraph as follows:

As discussed above, the invention can be implemented in any asset class or across asset classes. By way of example, the invention can be implemented in trading stock and equity options complex orders. The complex order process 50 [,] eliminates leg risk by executing such complex orders as a single transaction. The complex order process 50 records and processes market data from each of the markets on which the instruments are traded. According to one embodiment it receives information from SIAC (for CTS and CQS) and ISE (for ISE market and order information) via the interface 23. The complex order process 50 then processes the stock and options market information to determine matching opportunities and execute the legs of the trades in both markets as a single transaction at a net price. If the complex order process 50 cannot complete each leg of the complex order in a quantity consistent with any ~~ratios~~ ratio specified in the complex order, the complex order process 50 will not execute the trade. The complex order process 50 triggers may include additional events based on the particular type of asset class. According to one embodiment, when trading stock and equity option complex orders the complex order process 50 triggers on an updated BBO on the stock leg, changes to up-tick when short-selling certain securities, and the like. The complex order process 50 continues to

-12-

trade such complex orders at a single, net price, and ~~would further enhance~~ <u>enhances</u> liquidity available to these orders as a result of the wider price range available to it to execute such orders as a single transaction. According to this embodiment, the complex order process 50 calculates an options price range, a stock price range, and a total price range within which both the stock and options legs could trade.